UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/776,024	02/10/2004	Andrea Pagni	851763.447	9064

38106 7590 04/19/2006

SEED INTELLECTUAL PROPERTY LAW GROUP PLLC
701 FIFTH AVENUE, SUITE 6300
SEATTLE, WA 98104-7092

EXAMINER	
COLEMAN, ERIC	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 04/19/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/776,024	PAGNI ET AL.	
	Examiner	Art Unit	
	Eric Coleman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-24 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-16 and 19-24 is/are rejected.

7)☒ Claim(s) 17-18 is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☒ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 101

1. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Claim 24 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Claim 24 is directed to a computer program product that comprises software code portions of a process to performed on a computer that is directly loadable into a memory of a digital computer. The computer program product is does not constitute any invention of the statutory classes of invention (i.e., method, machine, manufacture, composition of matter). (Note claim is to the program product and not to the process). The feature of being directly loadable does not provide for the invention becoming a statutory class. The mere being able to take an abstract idea (of software code portions of a process) and inputting the abstract information directly into a memory such as by using a conventional input device does not provide for an invention that is with any of the statutory classes of invention.

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-16,19-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke (patent No. 6,934,832) in view of Schlansker (patent No. 6,408,428).

5. Van Dyke taught the invention substantially as claimed including a data processing ("DP") system comprising (as per claims 1, 24): providing a first set of registers (x86 set of registers) and providing a second set of registers (Tapestry physical machine registers) and mapping each of said first register set in a corresponding register of said second set designed to emulate the behavior or the register of said first set, performing an unique independent translation of the instructions of the first set into said second set (e.g., see col. 34, line 6-col. 35, line 67 and col. 32, line 1-65).

6. Van Dyke taught the operations providing a second set of registers and of mapping being obtained by adding functional units (x86 aligner and x86 converter)(to the processor and keeping the core unaltered) (e.g., fig. 1C)[The core contains a portion that performs native instructions and the difference between the system with the capability of performing emulated x86 instruction is the addition of the x86 converter and x86 aligner in figure 1c therefore the core for executing native instructions is left unchanged]. Further the Van Dyke x86 converter provided translation of the x86 instructions without direct access to the resources of the core (e.g., see fig. 1C).

7. Van Dyke taught translation x86 instruction into a native instruction set for the reason that x86 code can run slower than native code (e.g., see col. 32, lines 1-20). Van Dyke however did not expressly detail the native code comprised VLIW. Borrill however taught a dynamic decode unit for translating a complex instruction set in to native VLIW instructions (e.g., see col. 5, lines 19-39). Borrill also x86 instructions were the type of instructions being translated into a second native instruction set (e.g., see col. 1, lines

14-62). [this provides for an embodiment where instructions of an x86 instruction set is translated into a native VLIW instruction set].

8. One of ordinary skill in the DP art would have been motivated to combine the teachings of Van Dyke and Borrill. Both references were directed toward the problems of translating non-native instruction into native instructions for execution (e.g., see col. 1, lines 8-33 and col. 1, lines 38-67 of Borrill). Therefore one of ordinary skill would have been motivated to incorporate the Borrill teaching of translating complex instructions to VLIW instructions at least to take advantage of the efficiency of executing operations VLIW instructions and the feature of use of VLIW instruction to provide a reduced use of processor bandwidth (e.g., see col. 7, lines 32-41).

9. As per claim 20 Van Dyke taught a translation subsystem designed to receive at input an instruction of the first instruction set and supply at output a translation comprising one or more instructions of the second instruction set (e.g., see col. 34, line 6-col. 35, line 67 and col. 32, line 1-65) a translation memory coupled to the translation subsystem and structured to store said translation (native registers)(e.g., see col. 34, line 6-col. 35, line 67 and col. 32, line 1-65);control device for taking said translation from said translation memory and supplying it to the core of the processor (Instruction and decode dispatch)(e.g., see fig. 1C).

10. Van Dyke taught translation x86 instruction into a native instruction set for the reason that x86 code can run slower that native code (e.g., see col. 32, lines 1-20). Van Dyke however did not expressly detail the native code comprised VLIW. Borrill however taught a dynamic decode unit for translating a complex instruction set in to native VLIW

instructions (e.g., see col. 5, lines 19-39). Borrill also x86 instructions were the type of instructions being translated into a second native instruction set (e.g., see col. 1, lines 14-62). [this provides for an embodiment where instructions of an x86 instruction set is translated into a native VLIW instruction set].

11. As per claim 2, 13, Borrill taught adding plural functional units comprising adding translation units (210,220,230,240,250,260,270) external to the core of the native VLIW processor the translation device intercepting accesses to a storage area reserved to the instructions of the first instruction set(e.g., see fig. 5 and col. 4, lines 15-30 and see col. 1, lines 14-62 and col. 5, lines 8-65) [incoming instruction that would be stored instruction memory are intercepted and translated to VLIW instructions].

12. As to the limitations of claims 14,15,16 Borrill taught a control set of pointers in table 1 (e.g., see col. 30 lines 5-62) and since the pointers are separately accessed for comparision by the router then it would have been obvious to one of ordinary skill that there would have been in separate registers (e.g., see fig. 5). Further the converters are inactive until the router sends instruction for translation. Therefore one of ordinary skill would have been motivated to only activate the translators upon receipt of an appropriate instruction to save power (e.g., see fig. 4).

13. As per claim 3, Since Borrill taught processing an incoming instruction comprising determining the type of incoming instruction and determining the appropriate translation device for the instruction before decoding the instruction for execution (e.g., see fig. 4) when the processor had finished previous instruction and was waiting for a next instruction to be decoded by the appropriate decoder one of ordinary skill would

have been motivated to force the program counter (which indicates the next instruction to be executed) to point to the translation memory reserved in the translation device for containing a translation of the instruction belonging to the first (native instruction set) that is in the meantime decoded.[The waiting would have at least occurred because at that time the processor would not have had any instruction to process until the next instruction was decoded]

14. As per claim 4,Borrill taught loading in the reserved translation memory (e.g., see fig. 4) all instructions that constitute the translation of the decoded instruction of the set (e.g., see fig. 6, and col. 6, line 48-col. 7 line 32),

15. As per claim 5,23 further since Borrill taught plural converters for different type instructions and a router for routing to the particular decoder one of ordinary skill would have been motivated to provide a jump to link for jumping to the address of the converter process (e.g., see fig.3,4,5 and col. 4, lines 15-29) Further, Borrill taught a separate coding for the different types of instructions (e.g., see col. 4, lines 30-58). Therefore one of ordinary skill would have been motivated to use these separate coding identifiers to address the different decoder by using different addresses or pointers.

16. As to the Claim 6, Borrill taught translating all instructions of the first set to which there does not correspond to an equivalent single instruction in the second instruction set with an unconditioned GOTO jump (132,134)(e.g., see fig. 4)[if the system does not have an appropriate converter the system traps and emulates the instruction].

17. As per claim 7,8,Borrill taught mapping the registers of non-native to native registers for emulation of non-native instructions (x86 instructions). Also, Borrill taught a

separate coding for identifying the different types of instructions (e.g., see col. 4, lines 30-58)(this would have corresponded to the address of the instruction to be decoded). Therefore it would have been obvious to one of ordinary skill in the system of Van Dyke and Borrill the program counter register would have been emulated (e.g., see col. 34, line 3-col. 35, line 67 of Van Dyke) and the program counter would have been forced to the same value as for the VLIW processor and the counter of the emulated processor.

18. As to claim 9, since the Borrill reference taught different numerical idendifiers for different converters for native code emulating non-native code then it would have been obvious to one of ordinary skill that when the emulated code was finished an the system returned to the native code instructions then at times it would require a jump to the native code sequence (especially since there were plural different converters (DDU) for converting the code).

19. As per claims 10-12, The incrementing of an program counter for processing instruction that were emulated was conventional for sequential access to the next address of a sequence of instructions. Further since, as discussed above, with a jump to the sequence of native instructions one of ordinary skill would have been motivated to decrement the program counter by a predetermined amount to address the native instructions when the native instruction block was at a smaller address region than the emulated instructions. The incrementing or decrementing by a multiple of two (such as four or eight) was well known in the art with respect to computers such as Van Dyke and Borrill that would have addressed blocks or regions by addresses in a multiple of

two as the addresses are translated into binary for processing by the computer (e.g., see col. 3, line 46-col. 4 line 58 of Borrill).

20. As per claim 19, The reading of status flags for determining the state of the system was well known in the art at the time of the claimed invention. Also Van Dyke taught (e.g., see col. 38 lines 7-21) the performing a precheck to determine if the execution is about to enter a region of code that cannot be trusted to execute correctly.

21. As to claim 21, Van Dyke taught a translation table (e.g,. see figs. 9e,9c). As per claim 22, Borrill taught the converters between the input of instructions and the vliw functional units. Borrill taught memory (40) and one or more storage media would be coupled to the CPU (e.g., see col. 3, lines 35-45). One of ordinary skill would have been motivated to use a fast small memory or cache to provide fast access to data and instructions input to the processor as was well known in the art at the time of the claimed invention.

Allowable Subject Matter

Claims 17-18 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Moreno (patent No. 5,951,674) disclosed a object code compatible representation of very long instruction word programs(e.g,. see abstract).

Lin (patent No. 5,852,726) disclosed a system for executing two types of instructions that specify registers of a shared logical register file (e.g., see abstract).

Tremblay (patent No. 5,925,123) disclosed a processor for executing instruction sets received from a network or from local memory (e.g., see abstract).

Lin (patent No. 5,852,726) disclosed a system for executing two types of instructions that specify registers of a shared register file (e.g., see abstract).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

ERIC COLEMAN
PRIMARY EXAMINER